

Attorney Docket No. 36321.8009

Patent

Transmittal of Utility Patent Application for Filing

Certification Under 37 C.F.R. §1.10 (if applicable)

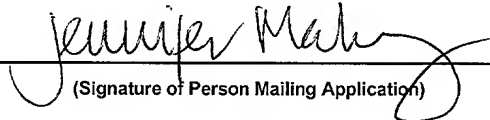
EL 889 537 109 US
"Express Mail" Label Number

January 2, 2001
Date of Deposit

I hereby certify that this application, and any other documents referred to as enclosed herein are being deposited in an envelope with the United States Postal Service "Express Mail Post Office to Addressee" service under 37CFR §1.10 on the date indicated above and addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231

Jennifer L. Mahoney

(Print Name of Person Mailing Application)


(Signature of Person Mailing Application)

METHOD AND APPARATUS FOR TRANSPARENT ENCRYPTION

5 **RELATED APPLICATIONS**

This application claims the benefit of United States Patent Application Numbers 60/259,754 and 60/259,786 filed January 4, 2001, 09/877,302 and 09/877,655 filed June 8, 2001, and 09/901,350 filed July 9, 2001, all of which are currently pending.

10 **TECHNICAL FIELD**

The claimed invention relates to the field of data security. In particular, the claimed invention relates to securing sensitive user data in a server system.

BACKGROUND

15 World Wide Web sites, or web sites, dealing with secure content use various mechanisms to protect this content. For example, electronic commerce, or e-commerce, web sites use a variety of mechanisms to protect user credit card numbers and user passwords. Most often, these sites use the Secure Socket Layer

(SSL) protocol to protect all sensitive data while it is in transit on the Internet among customer computers and browsers and the web site.

The SSL is a typical security protocol used on the web. The SSL protects data while it is in the network by encrypting it using a session-key known only to the web server and the client computer. The data is decrypted as soon as it reaches the web server. The web server processes the data (e.g., validating the credit card number) and then often stores it in a server database.

Unfortunately, however, many web servers store sensitive data in the clear, or in an unencrypted state, in an associated server database. As a result, this database is a prime target for hackers. Hackers have broken into web server databases, thereby compromising many credit card numbers and private user/customer information. These compromises are expensive for both electronic retailers and their customers.

While SSL protects transitory data in the network, it does not protect data once it reaches a web site and while it resides on the associated web servers. A different architecture is needed to protect data at the server site. Indeed, web sites should ensure that sensitive data stored in their database is always encrypted. However, any such system must permit efficient communication and not create bottlenecks that will annoy or discourage users of the network. If a security system does create bottlenecks, it could discourage or divert customers from the web site.

BRIEF DESCRIPTION OF THE FIGURES

The accompanying figures illustrate embodiments of the claimed invention. In the figures:

Figure 1 is a block diagram of a system architecture including a Transparent Encryption Appliance, under one embodiment.

Figure 2 is a block diagram of a system architecture including Transparent Encryption Appliances, under an alternate embodiment.

Figure 3 is a flow diagram of transparent encryption used in the embodiments.

Figure 4 is a block diagram of a system architecture 400 including oneTE Appliance 102 on a site front-end, and one TE Appliance on the site back-end, under an alternate embodiment.

5 **Figure 5** is a flow diagram of transparent encryption under an alternative embodiment using a public key.

Figure 6 is a flow diagram of transparent encryption under an alternative embodiment that protects user passwords against dictionary attacks.

10 In the drawings, the same reference numbers identify identical or substantially similar elements or acts. To easily identify the discussion of any particular element or act, the most significant digit or digits in a reference number refer to the Figure number in which that element is first introduced (e.g., element 108 is first introduced and discussed with respect to Figure 1).

15 Any headings used herein are for convenience only and do not affect the scope or meaning of the claimed invention.

DETAILED DESCRIPTION OF THE ILLUSTRATED EMBODIMENTS

20 A method and apparatus are provided for transparently protecting sensitive data within a server system or environment. Data entering and leaving a server site are evaluated for sensitive data. The sensitive data includes, e.g., credit card numbers and information, account numbers and information, and any other personal information of a customer or user that is of a sensitive nature, including birth date, social security number, and information related to user passwords. Upon detection of sensitive data, cryptographic operations are applied to the data. Cryptographic
25 operations include encrypting sensitive data transferred to the server system. Cryptographic operations also include decrypting encrypted sensitive data transferred from the server system en-route to a third party system. Further, cryptographic operations include hashing and keyed hashing of password data received at the server system. Moreover, cryptographic operations provide integrity for cookies.

The transparent protection is provided in an appliance of an embodiment that is separate from the server equipment. This appliance is coupled to the server systems and the data networks so that the server systems require no modification. In this manner, web site operators can install the appliances between their servers and the associated network connections without installing new hardware or software or modifying existing hardware or software on their servers.

In the description herein, numerous specific details are included to provide a thorough understanding of, and enabling description for, embodiments of the invention. One skilled in the relevant art, however, will recognize that the invention can be practiced without one or more of the specific details, or with other fields, expressions, methods, etc. In other instances, well-known structures or operations are not shown, or are not described in detail, to avoid obscuring aspects of the invention.

Unless described otherwise below, the construction and operation of the various blocks shown in Figures 1 and 2 are of conventional design. As a result, such blocks need not be described in further detail herein, because they will be understood by those skilled in the relevant art. Such further detail is omitted for brevity and so as not to obscure the detailed description of the invention. Any modifications necessary to these blocks, or other embodiments, can be readily made by one skilled in the relevant art based on the detailed description provided herein.

Each of the blocks depicted in the flowchart herein is of a type well known in the art, and can itself include a sequence of operations that need not be described herein. Indeed, unless described otherwise herein, the blocks depicted in the Figures are well known or described in detail in the above-noted and cross-referenced patent applications. Indeed, much of the detailed description provided herein is explicitly disclosed in the provisional patent applications; most or all of the additional material of aspects of the invention will be recognized by those skilled in the relevant art as being inherent in the detailed description provided in such provisional patent applications, or well known to those skilled in the relevant art. Those skilled in the

relevant art can implement aspects of the invention based on the flowchart of Figure 3 and the detailed description provided in the patent applications. For example, those skilled in the relevant art can create source code, microcode, program logic arrays or otherwise implement aspects of the invention based on these flowchart and the detailed description provided herein. Any routines may be stored in non-volatile memory (not shown) that forms part of an associated processor, or can be stored in removable media, such as disks, or hardwired or preprogrammed in chips, such as EEPROM or flash semiconductor memory chips.

Those skilled in the relevant art will appreciate that the routines and other functions and methods described herein can be performed by or distributed among any of the components described herein. While many of the embodiments are shown and described as being implemented in hardware (e.g., one or more integrated circuits designed specifically for a task), such embodiments could equally be implemented in software and be performed by one or more processors. Such software can be stored on any suitable computer-readable medium, such as microcode stored in a semiconductor chip, on a computer-readable disk, or downloaded from a server and stored locally at a client.

The Secure Socket Layer ("SSL") is a protocol that uses a significant amount of host server computing power. Many web sites use a security appliance or specially designed hardware device to manage the SSL traffic in order to offload some SSL work from the web servers. The security appliance is used coupled between the web server and the network connections, and handles the computations that support SSL connections.

A Transparent Encryption Appliance ("TE Appliance") is provided that provides backend database security at a web site, thereby protecting sensitive customer data stored and managed by the host web site systems and servers. The TE Appliance provides enhanced functionality in the form of transparent encryption to a security appliance.

Figure 1 is a block diagram of a system architecture 100 including a TE Appliance 102, under one embodiment. The TE Appliance 102 is coupled to a host web site or server system 104, and to numerous client computers and browsers 106 via at least one network 108. The network 108 includes the Internet as well as other wired, wireless, and hybrid network types, and may include independent networks, proprietary networks, or back plane networks, but is not so limited. Data transferred between the client computers 106 and the server system 104 passes through the TE Appliance 102. This includes both cleartext transactions, or Hypertext Transfer Protocol ("HTTP") transactions, and encrypted (SSL) transactions, as explained below. The TE Appliance may include a user interface 110. Also, the TE Appliance may include keys 120 of differing types.

Figure 2 is a block diagram of a system architecture 200 including Transparent Encryption Appliances 202 and 204, under an alternate embodiment. A first TE Appliance 202 is coupled to receive data transferred from numerous client computers and browsers (not shown) via the network 108. The received data is encrypted or hashed by the first TE Appliance 202, as appropriate to the type of data received, and the encrypted or hashed data is provided to the server system 104. As such, only encrypted or hashed data is available within the server system, in particular, in server system databases 210.

A second TE Appliance 204 is coupled to receive data transferred from the server system 104 to third party or other electronic systems (not shown) via the network 108. The data requested by the third party system is decrypted by the second TE Appliance 204.

The functionality provided by the TE Appliances of both embodiments can be hosted on dedicated network appliances as shown in **Figures 1 and 2**, but is not so limited. The transparent encryption functions can also be performed by, or distributed among any combination of, the host web site server systems 104 and 208, numerous client processing devices and browsers 106 coupled to the network, and any of the associated network components.

The TE Appliance of an embodiment can reside at the same physical location as the server systems that it supports or at different physical locations. Further, a TE Appliance may be configured to provide support to multiple server systems. It is also possible that the functions provided by the TE Appliance of an embodiment are distributed among numerous processing devices at numerous physical locations.

Figure 3 is a flow diagram of transparent encryption of both embodiments. In operation, the TE Appliance receives electronic transaction queries from client browsers and other electronic systems in block 302. The TE Appliance, in block 304, evaluates the requests entering the site. The evaluating or scanning functionality works with typical web encodings including Uniform Resource Identifier ("URI") encoding and Extensible Markup Language ("XML") encoding, but is not so limited. When the TE Appliance identifies tags indicating that the associated data is sensitive, it applies an appropriate cryptographic operation to the data within these tags, in block 306. For example, incoming sensitive data is encrypted using known encryption algorithms such as know public key infrastructure ("PKI") encryption algorithms or the Data Encryption Standard ("DES"). The resulting data is then, in block 308, routed to the appropriate component of the backend system or network.

The server environment, and the corresponding TE Appliances, also receive electronic information requests for sensitive data from third-party systems, in block 310, via network couplings with the third-party systems. For example, in the case of a purchase transaction, sensitive information including credit card information would have to be cleared with a financial institution before approving the purchase transaction. Upon receiving the request, encrypted sensitive data is retrieved and decrypted, in block 312. Once decrypted, the sensitive information is provided to the requesting third party in block 314, generally over a secure connection.

In an embodiment of the transparent encryption architecture, regular expressions are used to identify fields containing sensitive user information. For example, the regular expression "`^_.*`" is used to match any string that begins with

“__”, such as __password. Other forms of identifying sensitive fields, however, are also possible.

Transparent encryption can be applied to various messages in the HTTP protocol including, but not limited to, POST messages, GET messages, and HTML responses. The examples described herein illustrate the application of transparent encryption to HTML-encoded data. The same mechanism can be applied to other encodings, such as XML-encoded data, or data encoded in other formats, such as in other mark-up language formats.

An example application of transparent encryption includes POST ENCRYPT operations, wherein a POST body is received from a client of the form

```
&__password=mysecretpassword&__password_op=ENCRYPT&__password_key=bank_key&__password_rewrite=pwd”,
```

but is not so limited. The “__password” field portion supplies the user password. The “__” portion indicates that this field is to be processed by the TE Appliance. The field portion including “__password_op=ENCRYPT” indicates that the data in the “__password” fields is to be encrypted. The field portion including “__password_key=bank_key” provides the key name for use in encrypting the password. The field portion including “__password_rewrite=pwd” indicates that after transparent encryption processing, the field name changes from “__password” to “pwd”. Following transparent encryption processing the POST request body of an embodiment has the form “pwd=A124FFC9B306BI234AE”.

An example application of transparent encryption further includes POST DECRYPT operations, wherein a POST body is received from a client of the form

```
&__password=A124FFC9B306B1234AE&__password_op=DECRYPT&__password_rewrite=pwd”,
```

but is not so limited. The “__password” field portion supplies the user password. The “__” portion indicates that this field is to be processed by the TE Appliance. The field portion including “__password_op=DECRYPT” indicates that the data in the “__password” fields is to be decrypted. The field portion including “__

__password_rewrite=pwd" indicates that after transparent encryption processing, the field name changes from "__password" to "pwd". Following transparent encryption processing the POST body of an embodiment is of the form

"pwd=mysecretpassword". It is noted that there is no need for a "__password_key"

5 field since the cyphertext generated during the POST ENCRYPT operation identifies the key used to create the cyphertext.

An example application of transparent encryption also includes GET operations, wherein an original request is of the form

"GET /foo.html?__password=mysecretpassword&__

10 __password_op=ENCRYPT&__password_key=bank_key&__password_rewrite=password",

but is not so limited. The "__password" field portion supplies the user password.

The "__" portion indicates that this field is to be processed by the TE Appliance. The field portion including "__password_op=ENCRYPT" indicates that the data in the "__

15 __password" fields is to be encrypted. The field portion including "__

__password_key=bank_key" provides the key name for use in encrypting the

password. The field portion including "__password_rewrite=pwd" indicates that after transparent encryption processing, the field name changes from "__password" to

"pwd". Following transparent encryption processing the GET request body of an

20 embodiment has the form "GET /foo.html?password=A124FFC9B306B1234AE".

An example application of transparent encryption includes HTML responses, wherein a string similar to the POST request body is inserted inside an HTML

comment. The original response from a web server is of the form " credit card:

 <!--&__creditcard= A1CDF986FBC15456&__creditcard_op=DECRYPT&__

25 __creditcard_key=bank_key -->". It is noted that specification of an encryption key in

this original response example is not required as the cyphertext of an embodiment

may include encoded key identifiers. Following transparent encryption processing,

the HTML response is of the form "credit card: 1234 4567 1234 4567", but is not so limited.

The HTML files can be rather large files, so processing of these files may slow the TE Appliance. As such, the TE Appliance allows the administrator to restrict the URLs to which HTML response filtering is applied. Therefore, the administrator provides a list of regular expressions, and any URL matching any of these regular expressions that will be processed by the TE Appliance, such as the "___" expressions noted above.

Furthermore, the administrator can specify that transparent encryption processing should only be applied to a particular number, X, of bytes of the HTML file. The number X is generally on the order of 128 bytes indicating that all fields to which TE processing should be applied reside in the first 128 bytes of the HTML file, but the embodiment is not so limited. This value can be set at a large number indicating the entire HTML file is to be searched.

With reference to **Figure 1**, the TE Appliance 102 of an embodiment includes a user interface 110, but is not so limited. The user interface 110 enables the loading of symmetric encryption/decryption keys 120 onto the TE Appliance 102. It also enables the loading of public keys 120 onto the TE Appliance 102. Each key 120 is identified by a key-name.

The user interface also displays a fingerprint (hash) of all transparent encryption keys currently installed on the TE appliance. This enables a third party to apply the same hash function to the keys installed on the TE appliance, compare the hash result to previously computed and stored hash values for the stored key and verify that the correct keys are installed.

Moreover, the user interface enables a user or administrator to specify the list of fields to be processed by the TE Appliance. This is a list of regular expressions that identify Transparent Encryption fields. For example, setting "^_.*" as a delimiter implies that any field matching the regular expression "^_.*" is a Transparent Encryption field. For example, "__password" and "__creditcard" will be processed.

The user interface also allows an administrator to specify access controls to various keys installed on the module. For example, with reference to **Figure 2**, the administrator is able to specify that on TE Appliance 202 the key bank-key can only be used for encryption, while on TE Appliance 204 the bank-key can only be used for decryption. Thus, sites using two TE Appliances can specify that one TE Appliance is used for encryption, while the other is used for decryption.

Transparent encryption on a TE Appliance or web security appliance has many important applications. These applications include, but are not limited to, protecting credit card numbers/information, protecting sensitive user information, protecting passwords, providing integrity for cookies, and functioning as a key server.

Protecting sensitive user information, such as credit card numbers/information and bank account numbers/information, is a most natural application for transparent encryption. **Figure 4** is a block diagram of a system architecture 400 including one TE Appliance 102 on a site front-end, and one TE Appliance on the site back-end, under an alternate embodiment.

The front-end TE Appliance 102 of an embodiment is configured to inspect all requests entering the site via the network 108 and the client browsers 106. When a user request contains sensitive user data, the TE Appliance 102 is configured to encrypt the data using one of the installed keys 402. The server system 104 receives only the encrypted data. This encrypted data is stored in at least one database associated with the host web site 104.

The backend systems 404 connected to the server system 104 often need access to the sensitive user data. For example, with credit card numbers, the server system 104, or web site, often has to send the numbers to a financial clearing house 404 during the course of a transaction. Therefore, the server system 104 uses another TE Appliance 204 at the back-end. The back-end TE Appliance 204 of an embodiment is configured to use the installed keys 402 to decrypt all sensitive data passing through it enroute to the network 406 and back-end systems 404. This way, the credit card number is decrypted immediately before it is sent to the clearing

house 404. Again, none of the host web site internal systems 104 see the unencrypted credit card number. The network 406 can be a proprietary network, or can be the same type as network 108.

Figure 5 is a flow diagram of transparent encryption of an alternative embodiment using a public key. The TE Appliance of the alternative embodiment receives a request including a credit card number, at block 502. The TE Appliance identifies the sensitive information tags associated with the credit card number, at block 504, and encrypts the credit card number using an issuer's, or acquirer's, public key, under block 506. The web site sends the encrypted credit card number to the issuer, at block 508, and the issuer decrypts the number, at block 510. The issuer clears the transaction using the decrypted number. In this transaction, the web site never sees credit card numbers in the clear. This eliminates the risk of hackers breaking into the site and exposing customer credit card numbers.

Figure 6 is a flow diagram of transparent encryption of an alternative embodiment that protects user passwords against dictionary attacks. This function is realized when a front-end TE Appliance is configured to receive, under block 602, and inspect all user requests. When the TE Appliance detects a user password field, at block 604, it replaces the actual password "pwd" with a keyed hash function of the password $H_k(\text{pwd})$, under block 606. Any of a number of standard keyed hash functions (also known as message authentication code ("MAC") functions) can be used, for example HMAC-SHA1. The keyed hash function is stored in the server system, under block 608.

Referring to **Figure 1**, the hashing key 120 is preinstalled on the TE Appliance 102, but is not so limited. The hashed password is stored in the host web site 104 database. When a user logs in, the user provides the password as part of the user's request. The TE Appliance 102 detects the password and again applies the keyed hash function to the received password. The web site 104 then compares the hashed password to the value stored in the database, and authorizes the login if the two hashes match.

As a result of using this scheme, hackers that successfully break into the database only recover hashed passwords. Hashed passwords do not assist the hacker in logging into the site. Furthermore, the hacker is not able to mount an offline dictionary attack on the hashed passwords because the hacker does not have the key or keys used by the TE Appliance to hash the passwords. Hence, the TE Appliance prevents dictionary attacks on user passwords.

The TE Appliance of an embodiment also provides integrity for HTTP cookies. Typically, the HTTP cookies are used to store state on a user's web browser. The web site can send a cookie to the user and then retrieve the cookie from the user at a later time. Unfortunately, there is no mechanism for ensuring that users do not maliciously modify cookies while they reside on the user's machine. The TE Appliance can be used to overcome this problem.

When a web site sends a cookie to the user the TE Appliance appends a checksum or MAC to the cookie. When the user sends the cookie back to the site the TE Appliance can verify the checksum/MAC. If the checksum/MAC is not verified, the TE Appliance rejects the user's request. Otherwise, it forwards the user's request into the web site.

Web site administrators frequently place all secret keys on a single server called a key server. When a processing component of the site needs to apply cryptographic operations to data (e.g., encrypt, decrypt, or MAC), the processing component contacts the key server and requests that the key server perform this task. Currently there are no standard protocols for communicating with a key server. Each site implements a site-specific mechanism.

The TE Appliance of an embodiment functions as a key server. This is accomplished by installing the site's secret keys on the TE Appliance. The site's processing components or processors then issue standard HTTP requests to the TE Appliance in order to encrypt, decrypt, or MAC specified data. The response from the TE Appliance also uses the standard HTTP protocol. Hence, the TE Appliance is a convenient way for implementing a key server using standard web protocols.

Figures 1 and 2 and the discussion herein provide a brief, general description of a suitable computing environment in which aspects of the invention can be implemented. Although not required, embodiments of the invention are described in the general context of computer-executable instructions, such as routines executed
5 by a general purpose computer (e.g., a server or personal computer). Those skilled in the relevant art will appreciate that aspects of the invention can be practiced with other computer system configurations, including Internet appliances, hand-held devices, wearable computers, cellular or mobile phones, multi-processor systems, microprocessor-based or programmable consumer electronics, set-top boxes,
10 network PCs, mini-computers, mainframe computers and the like. Aspects of the invention can be embodied in a special purpose computer or data processor that is specifically programmed, configured or constructed to perform one or more of the computer-executable instructions explained in detail below. Indeed, the term "computer," as used generally herein, refers to any of the above devices, as well as
15 any data processor. Further, the term "processor" as generally used herein refers to any logic processing unit, such as one or more central processing units (CPUs), digital signal processors (DSPs), application-specific integrated circuits (ASIC), etc.

Aspects of the invention can also be practiced in distributed computing environments where certain tasks or modules are performed by remote processing
20 devices and which are linked through a communications network, such as a Local Area Network ("LAN"), Metropolitan Area Network ("MAN"), Wide Area Network ("WAN"), or the Internet. In a distributed computing environment, program modules or sub-routines may be located in both local and remote memory storage devices. Aspects of the invention described herein may be stored or distributed on computer-
25 readable media, including magnetic and optically readable and removable computer disks, hard-wired or preprogrammed in chips (e.g., EEPROM semiconductor chips), as well as distributed electronically over the Internet or over other networks (including wireless networks). Those skilled in the relevant art will recognize that portions of the invention reside on a server computer, while corresponding portions reside on a

client computer. Data structures and transmission of data particular to aspects of the invention are also encompassed within the scope of the invention. In general, while hardware platforms, such as the personal computers and remote computers, are described herein, aspects of the invention are equally applicable to nodes on a
5 network having corresponding resource locators to identify such nodes.

The elements and acts of the various embodiments described above can be combined to provide further embodiments. In general, alternatives and alternative embodiments described herein are substantially similar to previously described embodiments, and common elements and acts or functions are identified by the
10 same reference numbers. Only significant differences in construction or operation are described in detail.

One skilled in the relevant art will appreciate that the concepts of the invention can be used in various environments other than the Internet. For example, the concepts can be used in any electronic transaction environment. In general, a
15 display description may be in HTML format, email format or any other format suitable for displaying information (including character/code-based formats, algorithm-based formats (e.g., vector generated), and bitmapped formats). Also, various communication channels may be used, such as a local area network, metropolitan area network, wide area network, or a point-to-point dial-up connection instead of the
20 Internet. The server system may comprise any combination of hardware or software that can support these concepts. In particular, a web server may actually include multiple computers. A client system may comprise any combination of hardware and software that interacts with the server system. The client systems may include television-based systems, Internet appliances and various other consumer products
25 through which transactions may be conducted, such as wireless computers (palm-based, wearable, mobile phones, etc.).

Unless the context clearly requires otherwise, throughout the description and the claims, the words "comprise," "comprising," and the like are to be construed in an inclusive sense as opposed to an exclusive or exhaustive sense; that is to say, in a

sense of "including, but not limited to." Words using the singular or plural number also include the plural or singular number respectively. Additionally, the words "herein," "above," "below" and words of similar import, when used in this application, shall refer to this application as a whole and not to any particular portions of this application.

The description herein of illustrated embodiments of the invention is not intended to be exhaustive or to limit the invention to the precise form disclosed. While specific embodiments of, and examples for, the invention are described herein for illustrative purposes, various equivalent modifications are possible within the scope of the invention, as those skilled in the relevant art will recognize.

All of the references and U.S. patents and applications referenced herein are incorporated herein by reference. Aspects of the invention can be modified, if necessary, to employ the systems, functions and concepts of the various patents and applications described herein to provide yet further embodiments of the invention.

These and other changes can be made to the invention in light of the detailed description herein. In general, in the following claims, the terms used should not be construed to limit the invention to the specific embodiments disclosed in the specification and the claims, but should be construed to include all electronic systems that operate under the claims to provide secure electronic transactions. Accordingly, the invention is not limited by the disclosure, but instead the scope of the invention is to be determined entirely by the claims.

While certain aspects of the invention are presented below in certain claim forms, the inventors contemplate the various aspects of the invention in any number of claim forms. For example, while only one aspect of the invention is recited as embodied in a computer-readable medium, other aspects may likewise be embodied in a computer-readable medium. Accordingly, the inventors reserve the right to add additional claims after filing the application to pursue such additional claim forms for other aspects of the invention.